文章编号:1000-8055(2024)07-20220420-12

doi: 10.13224/j.cnki.jasp.20220420

基于改进花授粉算法的航空发动机 装配总体规划

章 斌1, 卢洪义1, 宋汉强2, 刘 舜1, 杨禹成1, 桑豆豆1

- (1. 南昌航空大学飞行器工程学院,南昌 330063;
 - 2. 海军研究院 上海研究所, 上海 200000)

要:针对航空发动机结构复杂、零件数量多且装配效率低、装配成本高的问题,提出了一种改进花 授粉算法(improved flower pollination algorithm, IFPA)的装配顺序优化方法。以装配优先性、装配稳定性、装配聚合性、装配重定向性和基础部件位置为影响因子构建优化目标评价体系,采用了不同的表示方案、反对立 学习的初始种群生成、动态调整的转换概率,在全局授粉和局部授粉规则中引入了均匀变异和精英变异,并加入遗传突变。运用在航空发动机低压压气机装配规划上,验证了 IFPA 的有效性,并讨论了 IFPA 的参数影响,并同粒子群算法、遗传算法、蚁群算法和花授粉算法进行比较,该算法找到最优序列的概率分别提高了41%、42%、41% 和 20%, 验证了 IFPA 在求解装配序列规划问题上的优越性。

关键词: 航空发动机压气机; 装配序列规划; 花授粉算法; 均匀变异; 精英变异

中图分类号: V263.2; TP391.9

文献标志码: A

Overall planning of aero-engine assembly based on improved flower pollination algorithm

ZHANG Bin¹, LU Hongyi¹, SONG Hanqiang², LIU Shun¹, YANG Yucheng¹, SANG Doudou¹

(1. School of Aircraft Engineering,

Nanchang Hangkong University, Nanchang 330063, China;

2. Shanghai Research Institute, Naval Academy, Shanghai 200000, China)

Abstract: In view of the problems of complex structure, large number of parts, low assembly efficiency and high assembly cost of aero-engine, an assembly sequence optimization method based on improved flower pollination algorithm (IFPA) was proposed. The optimization target evaluation system was constructed with the influence factors of assembly priority, assembly stability, assembly aggregation, assembly redirection and basic component position. Different representation schemes, initial population generation against independent learning, and dynamically adjusted transition probability were adopted, uniform and elite variation was introduced in global and local pollination rules, and genetic mutation was added. The effectiveness of IFPA was verified by applying it to the assembly planning of aero-engine low-pressure compressor, and the parameter influence of IFPA was discussed. And compared with particle swarm algorithm, genetic algorithm, ant colony algorithm and flower pollination algorithm, the probability

收稿日期: 2022-06-13

基金项目: 江西省自然科学基金(20201BBE51002); 江西省研究生创新专项资金项目(YC2021-S685)

作者简介:章斌(1998-),男,硕士生,主要从事航空发动机数字装配的研究。

通信作者: 卢洪义(1965-), 男, 教授、博士生导师, 博士, 主要从事发动机智能设计的研究。E-mail: 13964508115@163.com

引用格式:章斌, 卢洪义, 宋汉强, 等. 基于改进花授粉算法的航空发动机装配总体规划[J]. 航空动力学报, 2024, 39(7): 20220420. ZHANG Bin, LU Hongyi, SONG Hanqiang, et al. Overall planning of aero-engine assembly based on improved flower pollination algorithm[J]. Journal of Aerospace Power, 2024, 39(7): 20220420.

of finding the optimal sequence increased by 41%, 42%, 41% and 20%, respectively, which verified that IFPA can solve the assembly sequence planning superiority in question.

Keywords: aero-engine compressor; assembly sequence planning; flower pollination algorithm; uniform mutation; elite mutation

航空发动机的结构复杂,不同于普通机械 部件,从结构出发按部件起的作用和功能划分 为压气机、燃烧室、涡轮、附件传动机匣、减速 器、加力燃烧室等结构组件,从装配角度航空 发动机结构系统划分为零件、组合件、部件、单 元体、成品件和附件几大部分。在产品的制造 过程中装配是非常重要的,其结果对产品最终 的性能和质量影响重大[1]。航空发动机零件数 以万记,装配任务通常占生产成本和时间的很 大一部分。装配成本将花费 20%~50% 的制造 时间,40%以上的制造成本[2]。在航空发动机的 制造过程中提高装配效率、降低装配成本非常 的重要, 合理的装配序列会使航空发动机的装 配质量提高、生产周期减和制造成本降低[3]。装 配的序列规划(assembly sequence planning, ASP) 对于装配过程中生成令人满意的装配序列起着 重要作用。

ASP 是一个非确定性多项式(nondeterministic polynominal, NP)问题,目的是使用各种方法和 技术来寻找最佳装配序列。常用的方法是基于图 和基于树的搜索方法寻找最佳装配序列[4],但是 在装配零件呈指数增加时,通常陷入组合爆炸搜 索空间,难以选择出最优序列。随着国内外学者 对群体优化算法的研究应用,大量研究将一些优 化算法引入了实际工程的装配优化中,如蚁群算 法、果蝇优化算法、粒子群算法、烟花算法、遗传 算法、自适应引力搜索算法、模拟退火算法和花 授粉算法等[5]。郝博等[6]提出一种基于改进遗传 算法的飞机机翼装配序列规划方法,提出了非正 交干涉矩阵作为约束条件,通过改进遗传算法的 交叉和变异模式对其进行求解。最后以机翼为实 例验证。黄丰云等[7] 为了提高复杂产品的装配效 率,提出了遗传帝国竞争混合算法,给出了混合 算法的融合策略,用车后桥减速器为实例试验了 算法的有效性。王丰产等[8] 提出一种多工位多目 标装配序列优化的粒子群算法,提出了复杂多工 位产品的装配序列优化方法,最后以飞机起落架 为实例验证所提的多工位装配算法的有效性。刘 江伟等[9] 提出一种基于改进粒子群算法的多工位 装配序列规划方法,定义了粒子相似度和相似度 阈值,以相似度阈值控制变异的方法,提高了算 法的全局搜索能力。Wang 等[10] 提出了一种基于 蚁群算法的装配序列优化方法,利用干扰矩阵保证几何可行性,适应度函数由拆卸方向变化次数 组成。

Lu等[11]提出了一种利用蚁群算法求解集成 装配序列规划和装配线平衡的方法。基于搜索机 制和信息素更新机制,同时进行装配序列规划和 装配线平衡,以获得给定周期时间内的最优解。 Zhang 等[12] 提出了一种基于双种群搜索机制的离 散萤火虫算法用于装配序列规划。Murali等[13] 提出了一种新型的混合人工智能技术,该技术结 合模拟退火算法和人工免疫算法, 所提出的算法 旨在通过将人工免疫算法作为局部搜索策略来增 强模拟退火算法的性能。Li 等[14] 提出一种基于 和谐搜索算法的有效装配规划算法,该算法具有 出色的全局搜索能力,可以更有效地获得全局最 优。Li 等[15] 研究了一种基于烟花算法的装配序 列规划方法,研究了装配序列规划中烟花算法的 步骤,以某型飞机机翼装配为例,验证了所提方 法的有效性。Gao 等[16] 提出一种基于自适应引 力搜索算法(AGSA)的装配序列规划方法,构建 了算法优化的数学模型和基于 ASP 问题的组装 产品的编码,通过实例验证了所提方法的有效性。 Yang 等[17] 开发了花授粉算法(flower pollination algorithm, FPA), 采用转换概率在全局和局部搜 索之间转换,有效解决了全局和局部搜索的平衡。 Mishra 等[18] 开发了一种基于花授粉算法应用的 装配序列优化方法,首次应用于装配序列优化等 离散优化问题,在多模态优化问题方面表现更好。 Mishra 等[19] 提出一种改进的花授粉算法和混合 遗传花授粉算法,采用不同的表示方案、初始种 群生成策略以及局部和全局授粉规则的修改。上 述研究中采用了不同的优化算法对飞机部件、卫 星等产品进行装配序列规划,对于航空发动机装 配总体规划的研究很少。

本文提出了一种改进花授粉算法用于航空发动机装配总体规划。采用基于反对立学习的初始

种群生成、动态的转换概率,并在全局授粉中引入了均匀变异机制,扩大搜索空间。在局部授粉中引入精英变异策略,提高算法的寻优效率和收敛速度,在进入下一次迭代之前,利用遗传算法中的突变,避免算法早熟。本文以航空发动机的一个部件低压压气机为例,验证了算法的有效性,并得到了最优装配序列,同时分析对比了本文算法与粒子群算法、遗传算法、蚁群算法和文献[19]花授粉算法的性能,发现IFPA在收敛速度和寻最优方面明显优于其他算法。

1 航空发动机装配模型及适度函数

航空发动机是一种高度复杂和精密的装配体,工序繁多,零件数以万计,结构及其复杂。发动机外部管路、附件多数集中在局部,如图 1 所示,其装配过程更加烦琐,装配效率低。航空发动机的结构由许多不同的部件组成,如图 2 所示。其装配总体规划可以细分为各个部件的装配规划的组合,采用先各个组合部件的装配规划、后整体的装配规划的方法,对由n个零件组成的部件,每个零件由 p_i (i=1,2,…,n)表示。一个装配序列可以用S=[$p_1,p_2,…,p_n$]来表示。



图 1 航空发动机的外部管路、附件

Fig. 1 External piping and accessories of aero-engine



图 2 航空发动机核心机图

Fig. 2 Aero engine core diagram

1.1 装配几何可行性

零件之间在装配路径上的干涉关系可以通过 装配干涉矩阵来表示。建立零件在空间直角坐标 系中的干涉矩阵 *I*_m为:

$$I_{m} = \begin{bmatrix} I_{11}^{x} I_{11}^{y} I_{11}^{z} & I_{12}^{x} I_{12}^{y} I_{12}^{z} & \cdots & I_{1n}^{x} I_{1n}^{y} I_{1n}^{z} \\ I_{21}^{x} I_{21}^{y} I_{21}^{z} & I_{22}^{x} I_{22}^{y} I_{22}^{z} & \cdots & I_{2n}^{x} I_{2n}^{y} I_{2n}^{z} \\ \vdots & \vdots & \ddots & \vdots \\ I_{n1}^{x} I_{n1}^{y} I_{n1}^{z} & I_{n2}^{x} I_{n2}^{y} I_{n2}^{z} & \cdots & I_{nn}^{x} I_{nn}^{y} I_{nn}^{z} \end{bmatrix}$$
(1)

其中x、y、z为装配方向, I_{ij}^k 为零件i沿k(k=x,y,z)方向装配到位时与零件j的干涉情况,其取值为:

$$I_{ij}^{k} = \begin{cases} 1 & \$ 件 i \% k 方 向 装 配 时 与 \$ 件 j 发 生 干 涉 \\ 0 & \$ 件 i \% k 方 向 装 配 时 与 \$ 件 j 无 干 涉 \end{cases}$$
 (2)

装配序列的几何可行性用发生干涉的零件数目 f_{gs} 来描述。假设序列 $S = \{p_1, p_2, \cdots, p_{i-1}\}$ 为已装配序列, p_i 为待装配零件,定义 D_i 为零件 p_i 的可行装配方向集,则有

$$D_i = \sum_{i=1}^{i-1} I_{ij}^k \tag{3}$$

定义 G_i 为 p_i 发生干涉情况,其取值为:

$$G_i = \begin{cases} 1 & D_i \neq 0 \\ 0 & D_i = 0 \end{cases} \tag{4}$$

式中 $G_i = 0$ 表示零件 p_i 可装配, $G_i = 1$ 表表示零件 p_i 不能进行装配。

1.2 装配优先约束性

在可行的装配序列中,所有优先约束都会得到遵守,这些优先约束表示必须在该零部件之前装配的一组零部件。组件之间的优先约束通常由装配优先矩阵 $\mathbf{D} = [d_{ij}]_{n\times n}$ 表示,采用 d_{ij} 表示零件 p_i 与零件 p_i 的连接关系,其取值为:

$$d_{ij} = \begin{cases} 1 & \text{yn math } y \in \mathbb{R}, \text{math } y \in$$

且零件自身之间不存在优先约束关系,即当i = j时 $d_{ij} = 0$ 。装配序列的可行性程度根据序列中存在的可行性违规数量 f_{xx} 来衡量的,计算如下:

$$f_{yx} = \sum_{i=1}^{n-1} \sum_{i=1}^{i-1} d_{ji} \qquad 0 \leqslant f_{yx} \leqslant \frac{n(n-1)}{2}$$
 (6)

1.3 装配操作的稳定性

装配序列的稳定性是指零件或子装配体在参与装配过程中,保持装配零件稳定性的能力^[7]。

建立装配体的连接矩阵 $\mathbf{B} = (b_{ij})_{n \times n}$, 其中 b_{ij} 表示零件 p_i 与零件 p_i 的连接关系, 其取值为:

$$b_{ij} = \begin{cases} 2 & \$ \text{ ϕ_i} = \$ p_j$ 稳定连接 \\ 1 & \$ \text{ ϕ_j} = \$ p_j$ 连接, 夹具夹 p_j \\ 0 & \$ \text{ ϕ_i} = \$ p_j$ 没有连接关系 \end{cases}$$
 (7)

稳定连接表示零件之间通过紧固件或者是过 盈配合连接且零件自身之间不存在连接关系,即 当i = j时 $b_{ij} = 0$ 。

装配序列稳定性量化过程中,不仅要考虑当前装配零件与前一装配零件之间的连接关系,也考虑了与之前所有装配好的零件之间的连接关系[12],装配序列稳定性表示为:

$$f_{jc} = \sum_{i=1}^{n} L_i$$
 $0 \le f_{jc} \le 2n - 2$ (8)

其中 L_i 表示零件 p_i 的稳定性,如果 $b_{ij}(1 \le j \le i-1)$ 等于 $2, L_i = 2;$ 否则如果 $b_{ij}(1 \le j \le i-1)$ 中任何一个等于 $1, L_i = 1;$ 如果 $b_{ij}(1 \le j \le i-1)$ 都等于 $0, L_i = 0$ 。 f_{ic} 的值越大越稳定。

1.4 装配操作的聚合性

装配的聚合性表示装配过程中装配工具的改变次数,建立装配工具矩阵 $T = (t_i)_{n \times 1}$,装配工具的改变次数 f_{ni} 计算如下:

$$f_{gj} = \sum_{i=1}^{n-1} T_{si,i+1} \tag{9}$$

其中 $T_{s_{i,i+1}}$ 表示两个连续组装操作中组装工具的更改,其取值如下:

$$T_{\text{s}i,i+1} = \begin{cases} 0 & t_i = t_{i+1} \\ 1 & \text{其他} \end{cases}$$
 (10)

1.5 装配重定向性

装配重定向性即装配方向的改变次数,建立零件的装配方向矩阵 $\mathbf{C} = (c_i)_{n \times 1}$,装配方向改变次数 f_i 计算如下:

$$f_{\rm v} = \sum_{i=1}^{n-1} V_{i,i+1} \tag{11}$$

其中 $V_{i,i+1}$ 表示两个连续组装操作中装配方向的更改,其取值如下:

$$V_{i,i+1} = \begin{cases} 0 & c_i = c_{i+1} \\ 1 & \text{其他} \end{cases}$$
 (12)

1.6 装配中基础零部件的位置

装配序列中的基础零部件应是装配序列中的第一个零部件。基础零部件的位置值由 f_0 表示,其值计算如下:

$$f_b = \begin{cases} 0 & 基础零部件是第一个位置 \\ n & 其他 \end{cases}$$
 (13)

产品的基础零部件选择基于以下因素:它应该是所有部件中最重或最大的,应该与其他部件有最多数量的配合,应该易于固定和定位等[20]。

1.7 装配规划的适应度函数

以装配优先性、装配稳定性、装配聚合性、 装配重定向性和基础部件位置为影响因子构建优 化目标评价体系的适应度函数如式(14)所示,对 可行装配序列,设计了一个四个指标组成的函数 来衡量装配的成本。

$$F = f_{jc} + \omega_1 (2n - 2 - f_{yx}) + \omega_2 f_{gj} + \omega_3 f_v + \omega_4 f_b$$
(14)

其中 ω_1 、 ω_2 、 ω_3 、 ω_4 分别是各个指标的权重,为 $0\sim1$ 之间的实数,并且 $\omega_1+\omega_2+\omega_3+\omega_4=1$ 。

适应度值越小,表示装配序列成本越低,装配序列越优。

2 花授粉算法

花授粉算法最初由 Yang 等[17] 提出。在植物中,花通过授粉达到繁殖的目的,花授粉通常与花粉的转移有关,根据花朵授粉方式的不同分为异花授粉过程和自花授粉过程,授粉也可以采取自花授粉或异花授粉的形式[18]。异花授粉是指从不同植物的一朵花的花粉中进行的授粉,而自花授粉是指从同一朵花或同一植物的不同花朵的花粉中进行的一朵花的授粉。花授粉算法制定了以下规则:

- 1) 非生物自花授粉过程对应于花授粉算法 的局部搜索环节。
- 2) 生物异花授粉被视为全局授粉过程, 花粉 携带者进行 Levy 飞行。
- 3) 花的恒常性可以看作是繁殖概率, 正比于 两花相似性。
- 4) 局部授粉和全局授粉由转换概率 $p \in [0,1]$ 控制, 受地理位置和风等其他因素的影响, 局部 授粉在整个授粉活动中可能占相当大的比例。

基于上述的原则可以得到, 花授粉算法的操

作与实现主要通过生物异花授粉过程即全局搜索 环节和非生物自花授粉过程即局部搜索环节来 完成。

全局授粉(原则1)数学表示为:

$$\boldsymbol{X}_{i}^{t+1} = \boldsymbol{X}_{i}^{t} + \gamma L(\lambda) \left(\boldsymbol{G}^{*} - \boldsymbol{X}_{i}^{t}\right)$$
 (15)

其中 X_i 是迭代t时的花粉i的解向量; G^* 为当前群体最优解; γ 为控制移动步长的比例因子; $L(\lambda)$ 为授粉强度, 实质是与授粉强度相对应的基于Levy 的步长大小, 其值由 Levy 分布随机得到。 $L(\lambda) > 0$ 的 Levy 分布为:

$$L(\lambda) = \frac{\lambda \Gamma(\lambda) \sin(\pi \lambda/2)}{\pi} \cdot \frac{1}{s^{1+\lambda}}$$
 (16)

其中 $\Gamma(\lambda)$ 为标准伽马函数, Yang 曾建议 λ 的值为 1.5, 其中伽马函数值为 0.88。 s是利用 Mantegna 算法产生的步长:

$$s = \frac{\mu}{|v|^{1/\lambda}} \tag{17}$$

其中 μ 、 ν 服从正态分布, ν ~N(0,1), μ ~ $N(0,\sigma^2)$,方差 σ^2 计算方法为:

$$\sigma^2 = \left[\frac{\Gamma(1+\lambda)}{\lambda \Gamma(0.5+\lambda/2)} \cdot \frac{\sin(\pi \lambda/2)}{2^{\lambda/2-0.5}} \right]^{1/\lambda}$$
 (18)

局部授粉(规则2)可以用如下公式表示:

$$\boldsymbol{X}_{i}^{t+1} = \boldsymbol{X}_{i}^{t} + \varepsilon \left(\boldsymbol{X}_{i}^{t} - \boldsymbol{X}_{k}^{t} \right) \tag{19}$$

其中 X_j 、 X_k 是从第t次迭代解集中随机抽取的解向量, ε 是[0,1]内的均匀分布中的随机数。

对于平衡局部授粉与全局授粉活动的转换概率的取值问题,一般取 0.8 会得到比较好的效果[17]。

3 改进花授粉算法

一般的群智能算法存在着容易陷入局部最优、 收敛速度变慢、进化后期种群多样性丢失的问题, 基本 FPA 也存在一样的问题,为了使花授粉算法 避免这些问题。本文提出一种改进花授粉算法。

3.1 初始种群生成

FPA 中的初始种群是随机生成的,没有考虑这些序列的可行性。为扩大初始化解的范围,使用 Li 等[14] 在装配规划中引入的反对立学习方法。该方法能显著提高算法的性能。假设 $p(x_1, x_2, \dots, x_n)$ 是n维的解向量,且 $x_i \in [x^1, x^u]$, $i = 1, 2, \dots$, n, x^1 和 x^u 分别是变量 x_i 的下限和上限,在反对立

学习方法中,反向解向量 $p'(x'_1,x'_2,\cdots,x'_n)$ 定义如下:

$$x_i' = x^{l} + x^{u} - x_i \tag{20}$$

3.2 动态转换概率

由于参数对 FPA 性能的影响,对于不同类型的问题其取值不同,由 FPA 算法的规则可知转换概率取值为定值,若取值过大,则侧重于全局搜索,算法收敛速度变慢,影响算法的全局优化能力,若取值过小,算法容易陷入局部极值。为解决这个问题,对转换概率采用动态自适应调整的策略。

$$p(t) = p_{\min} + (p_{\max} - p_{\min}) \times \left(1 - \frac{t}{N_{\text{iter}}}\right)$$
 (21)

其中转换概率 $p \in [0.2,0.9]$, t为当前迭代次数, N_{iter} 为最大迭代次数, p_{\min} 和 p_{\max} 分别是参数p的最小值和最大值。

在进化迭代前期, p的取值大, 算法主要进行全局搜索, 搜索的范围更大, 随着迭代进一步深入, p取值逐步减小, 算法重点在于局部搜索, 搜索更精细, 有利于找到最优解。

3.3 均匀变异的离散全局授粉方式

花授粉算法在全局授粉中完全陷入局部最优, 无法跳出局部最优。采用均匀变异的策略调控花 粉的多样性,解决算法在陷入局部最优时无法跳 出的问题。改进的全局授粉方式如下:

$$X_i^{t+1} = X_i^t + \gamma L(\lambda) \left(G^* - X_i^t \right) + \sigma \left(X_i^t - X_k^t \right)$$
 (22)

其中 X_{j} 、 X_{k} 为从第t次迭代种群中随机选取的两个不同的个体, σ 是[0,1]服从均匀分布的随机数。

为使全局授粉能够处理离散问题,设定了以 下离散规则:

1) 装配序列 $p_i(x_1, x_2, \dots, x_n)$ 和 $p_j(x_1, x_2, \dots, x_n)$ 相减得到一个新装配序列 $p_k(x_1, x_2, \dots, x_n)$,考虑到结果会有负值,采用绝对值。计算公式如下:

$$x_k = \begin{cases} x_i - x_j & x_i > x_j \\ x_j - x_i & \text{其他} \end{cases}$$
 (23)

2) $L(\lambda)$ 是 Levy 分布随机生成的整数数组, $L(\lambda)$ 数组中的值由式(16)生成,这个数组值的特点是,在大多数时间生成一个小的数,突然地随机跳到一个非常大的数字,从而模仿鸟类的 Levy 飞行行为。 σ 是一个由随机生成的数字 0 和 1 组成的数组。

3) 数组相乘, 对应的数值相乘, 规则如下:

$$x_k = x_i x_i \tag{24}$$

4) 数组相加,对应的数值相加,规则如下:

 $(G^* = (5,10,8,16,14,13,15,7,12,3,17,18,11,4,2,1,6,9))$

 $X_i^t = (10,12,13,2,7,16,6,11,3,17,4,5,18,1,8,15,14,9)$

 $G^* - X_i^t = (5,2,5,14,7,3,9,4,9,14,13,13,7,3,6,14,8,0)$

 $L(\lambda) = (4,0,0,3,5,5,5,1,8,4,7,0,1,7,1,2,1,1)$

 $L(\lambda) (G^* - X_i^t) = (20,0,0,42,35,15,45,4,72,56,91,0,7,21,6,28,8,0,6,28,8,0)$

 $X_i^t = (0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17)$

 $X_k^t = (0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17)$

 $X_{i}^{t} - X_{k}^{t} = (5,9,6,13,10,8,9,0,4,6,7,7,1,9,12,14,10,8)$

 $\sigma = (1,1,0,0,0,0,1,1,1,1,1,0,0,1,1,1,0,1)$

 $\sigma(X_j^t - X_k^t) = (5,9,0,0,0,0,9,0,4,6,7,0,0,9,12,14,0,8)$

 $X_i^{t+1} = X_i^t + L(\lambda) (G^* - X_i^t) + \sigma(X_i^t - X_k^t) = (35,21,13,44,42,31,60,15,79,79,102,5,25,31,26,57,22,17)$

在得到的装配序列中有超过最大零件数的值, 对于这些数值用均匀分布随机生成的小于最大零 件数的随机数替换。例如35大于18,超过最大 零件数,被均匀分布的随机数值4替换掉,全部检 查后新的装配序列如下:

$$X_i^{t+1} = (4,3,13,6,5,9,11,15,13,11,7,5,17,12,6,5,12,17)$$
(27)

重复任何组件编号也会导致序列不可行。为 了使序列可行,基于优先约束,采用修复方法将 重复的部件编号替换为缺失的部件编号,修复步 骤如下:

- 1) 统计识别数组中重复的数值和缺失的数 值,并把数组中重复的数值变更为0。
- 2) 根据优先矩阵计算它们的优先数, 并根据 它们优先数的大小排序。
- 3) 按照排序的位置把数值按顺序填入数组 中为0的位置,得到新的序列。

全部修复后的新装配序列如下:

$$X_i^{t+1} = (4,3,13,14,6,9,5,15,16,12,7,17,11,18,10,8,2,1)$$
(28)

3.4 精英变异的离散局部授粉方式

花授粉算法在局部授粉中随机产生的子代能 够较好地保持算法种群个体之间的差异性,使算 法能够保持较好的持续优化能力,但会降低算法

 $x_k = x_i + x_i$

(25)

规则(其中以γ=1来示例),说明了上述离散化处理:

的收敛速度和局部搜索能力。针对上述问题,采 用精英变异策略,同时在局部搜索中设置转换概 率θ。精英策略使种群个体向局部最优个体的位 置进化,提升算法的收敛速度,提高了算法的寻 优性能,但会陷入局部最优。采取非线性递减概 率,使得算法在前期重点在全局搜索,在进化后 期重点在局部搜索,提高算法的寻优能力。局部 授粉规则如下:

如果 $N_{\text{rand}} > \theta$,则有

$$X_{i}^{t+1} = X_{\text{best}}^{t} + \alpha (X_{a}^{t} - X_{b}^{t}) + \beta (X_{m}^{t} - X_{n}^{t})$$
 (29)

否则

$$\boldsymbol{X}_{i}^{t+1} = \boldsymbol{X}_{i}^{t} + \delta \left(\boldsymbol{X}_{i}^{t} - \boldsymbol{X}_{k}^{t} \right) \tag{30}$$

其中 X_a^t 、 X_b^t 、 X_m^t 、 X_n^t 、 X_n^t 和 X_k^t 为第t次迭代种群 中随机选取的6个不同的个体, Nrand为[0,1]均匀 分布的随机数, X_{best}^t 是第 t 次迭代的局部最优解向 量, δ 、 α 和β是均值为 0.5、标准差为 0.1 的高斯分 布随机数, θ 为局部转换概率,计算公式如下:

$$\theta = 1 - \frac{t}{N_{\text{tor}}} \tag{31}$$

采用全局授粉规则中的离散化处理,将其应 用在局部授粉规则中,下面的例子说明了修改后 的局部授粉规则和步骤流程:

1) 如果 $N_{\text{rand}} > \theta$, 则

$$\begin{aligned} & X_a^t = (17,18,1,13,3,2,10,11,5,14,15,16,12,6,9,7,8,4) \\ & X_b^t = (5,15,11,6,1,2,17,7,16,4,18,14,13,3,10,12,9,8) \\ & X_a^t - X_b^t = (12,3,10,7,2,0,7,4,11,10,3,2,1,3,1,5,1,4) \\ & \alpha = (1,1,0,0,1,0,0,1,0,0,1,1,0,0,1,0,0,1) \\ & \alpha \left(X_a^t - X_b^t \right) = (12,3,0,0,2,0,0,4,0,0,3,2,0,0,1,0,0,4) \\ & X_m^t = (5,15,11,6,1,2,17,7,16,4,18,14,13,3,10,12,9,8) \\ & X_n^t = (5,16,13,3,14,8,17,1,18,11,2,7,10,12,9,6,4,15) \\ & X_m^t - X_n^t = (0,1,2,3,13,6,0,6,2,7,16,7,3,9,1,6,5,7) \\ & \beta = (0,1,1,1,0,0,1,0,1,0,1,1,1,0,0,1,0,1) \\ & \beta \left(X_m^t - X_n^t \right) = (0,1,2,3,0,0,0,0,2,0,16,7,3,0,0,6,0,7) \\ & X_{best}^t = (7,15,3,14,5,2,13,18,11,12,10,6,4,9,8,16,1,17) \\ & X_t^{t+1} = X_{best}^t + \alpha \left(X_a^t - X_b^t \right) + \beta \left(X_m^t - X_n^t \right) = (19,19,5,17,7,2,13,22,13,12,29,15,7,9,9,22,1,28) \end{aligned}$$

对于数组中超过最大零件数的数值,采用全局离散化中的随机数替换的方法,全部检查后新的装配序列如下:

 $X_i^{t+1} = (7,1,5,17,7,2,13,10,13,12,8,15,7,9,9,17,1,16)$ (33)

对于一些重复的部件编号,采用改进的全局

授粉规则中的使用基于优先矩阵的修复策略,进 行修复。

全部修复后的新装配序列如下:

$$X_i^{t+1} = (13,4,5,7,14,2,6,10,17,12,8,15,11,18,9,3,1,16)$$
(34)

2) 如果 $N_{\text{rand}} \leq \theta$, 则

$$\begin{cases} X_{j}^{t} = (13,16,4,6,12,9,7,17,14,11,8,18,10,2,3,15,5,1) \\ X_{k}^{t} = (17,14,13,4,7,15,12,6,18,11,5,10,16,8,3,2,9,1) \\ X_{j}^{t} - X_{k}^{t} = (4,2,9,2,5,6,5,11,4,0,3,8,6,6,0,13,4,0) \\ \delta = (0,1,1,1,1,1,1,0,1,1,0,1,1,1,1,1,0) \\ \delta (X_{j}^{t} - X_{k}^{t}) = (0,2,9,2,5,6,5,11,0,0,3,0,6,6,0,13,4,0) \\ X_{i}^{t} = (2,6,11,12,16,4,3,5,9,1,14,18,15,17,10,7,13,8) \\ X_{i}^{t+1} = X_{i}^{t} + \delta (X_{i}^{t} - X_{k}^{t}) = (2,8,20,14,21,10,8,16,9,1,17,18,21,23,10,20,17,8) \end{cases}$$

对于数组中超过最大零件数的数值,采用全局离散化中的随机数替换的方法,全部检查后新的装配序列如下:

$$X_i^{t+1} = (2, 8, 17, 14, 11, 10, 8, 16, 9, 1, 17, 18, 16, 12, 10, 8, 17, 8)$$
 (36)

对于一些重复的部件编号。采用改进的全局 授粉规则中的使用基于优先矩阵的修复策略,进 行修复,全部修复后的新装配序列如下:

$$X_i^{t+1} = (2,13,4,14,11,7,6,15,9,1,5,18,16,12,17,10,8,3)$$
(37)

3.5 基于变异的改进策略

从FPA的构成中知道FPA算法用式(21)得

到的参数 p 的值来决定选择全局或者局部授粉, 会产生更新的个体陷入局部最优的现象,影响 FPA 的寻优能力。本文利用遗传算法中的突变思想, 在算法进入下一次的迭代之前,进行突变变异来 拓展种群搜索空间,可以使算法跳出局部最优,保 持种群的多样性,增大寻优的能力。改进规则如下:

变异算子设定变异概率 p_b ,如果 \in [0,1]的随机数小于变异概率 p_b ,随机选择两个不同的位置,交换它们的值,重复交换5次,完成变异。否则,跳过。

3.6 改进花授粉算法实现步骤流程

根据上述的改进,提出一种改进花授粉算法。 改进花授粉算法步骤流程图如图 3 所示,算法流程步骤如下:

步骤 1 参数初始化, 确定种群数n, 维数(零

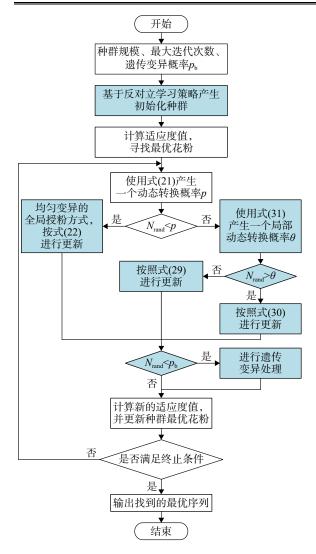


图 3 改进花授粉算法流程图

Fig. 3 Flowchart of improved flower pollination algorithm

件数)D,最大迭代次数 N_{iter} 。

步骤 2 采用基于反对立学习的初始种群生成,并计算出种群个体的适应度值,保存种群中的最优适应度值和对应的装配序列。

步骤 3 由式(21)产生一个转换概率 p, 若 $N_{\text{rand}} < p$ 转步骤 4, 否则转步骤 5。

步骤 4 全局授粉,按式(22)对种群个体进 行更新。

步骤 5 局部授粉, 由式(31)产生一个转换 概率 θ , 若 $N_{\text{rand}} > \theta$ 转步骤 6, 否则转步骤 7。

步骤 6 按式(29)对种群个体进行更新。

步骤7 按式(30)对种群个体进行更新。

步骤 8 若 $N_{\rm rand} < p_{\rm b}$ 则进行遗传变异, 否则跳过。

步骤 9 所有种群个体是否全部都更新, 否则转步骤 3。

步骤 10 计算种群更新的个体的适应度值,

寻找更新种群中最优适应度值和对应的解,并更 新最优适应度值和对应是装配序列。

步骤 11 判断是否满足结束条件, 是, 输出最优值和对应的解并退出程序, 否则转步骤 3。

4 实验验证与分析

4.1 改进花授粉算法实验验证

航空发动机的装配总体规划,由压气机、燃烧室、涡轮、附件传动机匣、减速器、加力燃烧室等部件的装配规划组成,为了验证算法的有效性,在 Visual Studio 2017 软件上使用 C++编程实现本文算法,运行环境如下: CPU 主频为 2.9 GHz, 内存为 16 GB, Windows10 64 位操作系统。本文以航空发动机中的低压压气机(如图 4 所示)为示例进行实验,该压气机共 18 个零件(编号如图 5 所示),装配体的基础零部件是零件 13,零件信息、装配工具和装配方向信息如表 1 所示。表中的装配工具 T1~T4 是指按安装零件使用的夹具和工具组合方案,根据零件安装的配合关系和零件的类型分成四类。由用户提供的装配优先矩阵 $D = [d_{ij}]_{nxn}$ 和连接矩阵 $B = (b_{ij})_{nxn}$ 如下:

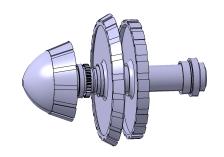


图 4 航空发动机低压压气机

Fig. 4 Aero engine low pressure compressor

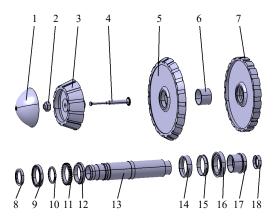


图 5 航空发动机低压压气机爆炸图

Fig. 5 Explosion diagram of aero engine low pressure compressor

(38)

(39)

对本文算法的模拟运行,参数设置如下: 初始种群n=10,遗传变异概率 $p_b=0.05$, $\lambda=1$,迭代次数 $N_{\text{iter}}=500$,假设适应度函数中的权重种群规模逐渐增加到 200,步长 γ 由 1增加到 18。获得的最优序列为(13, 14, 15, 16, 17, 7, 4, 6, 5, 18, 12, 11, 10, 9, 8, 3, 2, 1),序列中没有优先干涉,装配稳定性值为 20, 其具有最少的工具变化(即 7 次)和方向变

表 1 低压压气机装配信息

Table 1 Low pressure compressor assembly information

编号	零件名称	装配工具	方向
1	整流罩	Т3	+z
2	螺母	T4	$+_Z$
3	第1级压气机盘	Т3	+z
4	花键螺栓	Т3	$+_Z$
5	第2级压气机盘	T1	+z
6	间隔衬套	T1	+z
7	第3级压气机盘	T1	+z
8	前涨圈座	T2	+z
9	前轴承	T2	$+_Z$
10	前间隔衬套	T2	+z
11	主动齿轮	T2	$+_Z$
12	后涨圈座	T2	$+_Z$
13	低压压气机轴	T1	-z
14	中介支撑轴承螺帽	T4	-z
15	涨圈座	Т3	-z
16	前中介轴承	Т3	-z
17	中介轴承衬套	Т3	-z
18	螺母	T4	-z

化(即 4 次), 装配基础部件为零件 13, 该装配序列可行, 获得的最优适应度值为 8.35 (假设式(10)中行, 获得的最优适应度值为 8.35 (假设式(10)中给出的适应度函数中的权重 ω_1 = 0.4, ω_2 = ω_3 = 0.25, ω_4 = 0.1)。对于 500 次迭代, 提供改进花授粉最佳性能的一组参数, 在寻找全局最优解方面的一致性如下: 种群大小为 200, 改进遗传变异概率固定为 0.1, γ 步长为 9。对于上述参数集, 发现找到最优序列的一致性为 43%(基于改进花授粉算法模拟的 100 次独立运行)。图 6显示了改进花授粉算法在 500 次迭代中的收敛图。

影响性能的两个主要参数是种群大小和 IFPA 步长。为了检查上述参数对改进花授粉算法性能的影响,改变上述两个参数,每次模拟运行重复 100次,其结果如表 2 所示,都给出了每 100 次独

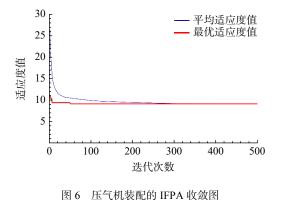


Fig. 6 IFPA convergence plot for compressor assembly

立运行的全局最优适应度值的平均值,在 100 次运行中找到最优适应度值的次数和最优适应度值。从表 2 的结果可以看出,改进花授粉算法参数的最优选择是种群大小 200, IFPA 步长为 9。

表 2 不同参数对 IFPA 的影响 Table 2 Effect of different parameters on IFPA

种群数	詳数 步长 平		次数	最优值
	1	9.502 5	1	8.85
10	9	9.495	1	8.85
	18	9.495	2	8.85
	1	9.435	4	8.85
20	9	9.417 5	5	8.85
	18	9.39	7	8.85
	1	9.32	12	8.85
50	9	9.327 5	11	8.85
	18	9.285	16	8.85
	1	9.23	24	8.85
80	9	9.15	20	8.85
	18	9.175	20	8.85
	1	9.12	31	8.85
150	9	9.075	37	8.85
	18	9.077 5	33	8.85
	1	9.09	39	8.85
200	9	9.04	43	8.35
	18	9.067 5	40	8.85

4.2 算法对比分析实验

将本文的改进花授粉算法(IFPA)与基本花授粉算法(FPA)、粒子群算法(PSO)、蚁群算法(ACO)和遗传算法(GA)进行比较,对上述算法分别进行了100次独立的500次迭代仿真。

对于粒子群算法,为获得最优适应度给出的参数设置是惯性权重 ω =0.9、加速因子 c_1 =2、 c_2 =2。对于遗传算法,为获得最优适应度给出的参数设置是交叉概率为0.90和变异概率为0.05。对于蚁群算法,为获得最优适应度给出的参数设置是信息素浓度权值 α =1、启发信息重要性因子 β =5和信息素挥发系数 ξ =0.1, ρ =0.2。对于文献[19]的花授粉算法,为获得最优适应度采用文献中给出的参数设置,转换概率为0.1、步长为9。对于本文改进花授粉算法,使用了上一节中给出的最佳参数,步长为9。为了保持结果比较的公平性,对于所有算法的模拟运行,种群大小和最大迭代次数分别保持在200和500。此外,所有算法使用了式(14)中给出的相同适应度函数。

表3总结了所有5种算法的比较结果,5种算法都是以航空发动机低压压气机为对象进行模拟仿真。对于每种算法,都给出了最优适应度值、最优适应度值的平均值、100次独立运行中的获得最优适应度值的次数,并给出了获得最优装配序列的概率。通过表3的数据对比分析可以发现,在这5种算法中,粒子群算法、文献[19]的花授粉算法和本文提出的改进花授粉算法找到了可行的最优装配序列,其适应度值分别为9.1、8.85和8.35。遗传算法和蚁群算法找到的最优序列对应的适应度值分别为13.75和13.35,且序列是不可行的。

表 3 5 种算法 100 次对比结果
Table 3 100 comparison results of 5 algorithms

参数	GA	PSO	ACO	FPA	IFPA
最优适应度值	13.75	9.1	13.35	8.85	8.35
最优适应度值平均值	18.95	10.96	18.31	9.23	9.04
最优序列次数	1	2	1	23	43
获最优的概率/%	1	2	1	23	43

改进花授粉算法可以在 100 次运行中找到 43 次最优装配序列, 使得在 500 次迭代中找到最优装配序列的概率达到 43%, 比文献 [19]FPA 找到最优序列的概率提高了 23%, 而比 GA、PSO 和 GA 找到最优序列的概率提高了 41%、42% 和 41%。

表 4 列出了通过 5 种算法得到的航空发动机 低压压气机的最优序列,并给出了对应的最优适 应度值、可行性违规数、装配稳定性数、装配工 具改变次数、重定向次数和基础部件值。5 种算 法的平均适应度值的收敛图和全局最优适应度值 的收敛图如图 7 和图 8 所示。

通过表 4 的数据对比分析可以发现,本文提出的改进花授粉算法给出的最优装配序列需要最少的装配工具改变次数(7次),最少的装配方向重定向次数(4次),且稳定性最高(为 20)。通过比较所有算法的全局最优适应度收敛图,发现改进花授粉算法明显优于其他 4 种算法,首先改进花授粉算法找到了最优的适应度值为 8.35,且在算法早期收敛速度最快,与其 PSO、GA 和 ACO相比,FPA 和 IFPA 迭代的收敛速度更快,且 IFPA 比 PFA 找到更优的适应度值,即 IFPA 找到更优的装配序列。这表明,本文提出的改进花授粉算法的性能优于其他 4 种算法,即改进花授粉算法的性能优于其他 4 种算法,即改进花授粉算法

算法	最优序列	最优 适应度值	可行性 违规数	装配稳定 性数	装配工具 改变次数	重定 向次数	基础 部件值
GA	(13,14,7,6,5,15,12,11,9,8, 4,16,3,10,2,1,17,18)	13.75	3	19	12	7	0
PSO	(13,14,15,16,7,6,5,12,17,11, 18,10,9,8,4,3,2,1)	9.1	0	20	10	4	0
ACO	(13,14,16,6,7,15,5,17,11,18, 12,10,9,8,4,3,2,1)	13.35	3	20	12	7	0
FPA	(13,7,6,4,5,12,14,15,16,11, 17,18,10,9,8,3,2,1)	8.85	0	20	9	4	0
IFPA	(13,14,15,16,17,7,4,6,5,18, 12,11,10,9,8,3,2,1)	8.35	0	20	7	4	0

表 4 5 种算法获得的压气机最佳序列比较 Table 4 Comparison of compressor optimal sequences obtained by five algorithms

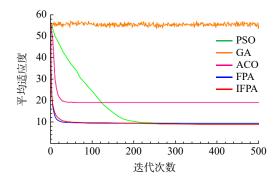


图 7 5 种算法平均适应度值收敛图 Fig. 7 Convergence graph of average fitness value of 5 algorithms

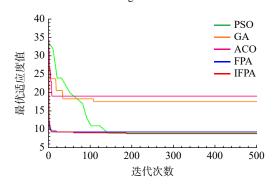


图 8 5 种算法最优适应度值收敛图 Fig. 8 Convergence graph of optimal fitness value of 5 algorithms

和 AG 相比具有更快的收敛速度,并且具有更强的鲁棒性。

5 结 论

针对航空发动机的装配总体规划产品问题,本文提出了一种改进花授粉算法的装配序列优化方法,构建了由装配优先性、装配稳定性、装配聚合、装配重定向性和基础部件位置5个指标相关的可行装配序列的适应度函数。在种群初始化中采用了反对立学习方法,提高了IFPA的性能,改

进了动态转换概率,在全局授粉规则中引入了均 匀变异的策略,在局部授粉规则引入了精英变异 的策略,提高了算法的寻优效率,使其收敛速度 得到提升,在进入下一次迭代之前,利用遗传算 法中的突变,来拓展种群搜索空间,保持种群的 多样性,增大寻优的能力。对于航空发动机的装 配总体规划,以航空发动机低压压气机为示例, 验证了算法有效性。并讨论了 IFPA 的参数影响, 每次模拟运行重复100次,找到了最佳参数。并 将 IFPA 与 PSO、GA、AG 和文献 [19] 中的 FPA 算法结果进行了比较,通过对比分析发现,提出 的 IFPA 在找到最优序列的收敛速度明显高于其 他算法,找到最优解方面明显优于其他算法。本 文算法可应用在航空发动机其他部件的装配规划 中,在航空发动机的总体规划中能够提高装配效 率,降低装配成本。

参考文献:

- [1] 朱梅玉,李梦奇,文学,等. 汽轮机转子动叶片装配序列智能优化[J]. 航空动力学报, 2017, 32(10): 2536-2543.

 ZHU Meiyu, LI Mengqi, WEN Xue, et al. Intelligent optimization of turbine rotor blade assembly sequence[J]. Journal of Aerospace Power, 2017, 32(10): 2536-2543. (in Chinese)
- [2] GAO Shang, JIN Ruoyu, LU Weisheng. Design for manufacture and assembly in construction: a review[J]. Building Research & Information, 2020, 48(5): 538-550.
- [3] DEEPAK B, BALA MURALI G, BAHUBALENDRUNI M R, et al. Assembly sequence planning using soft computing methods: a review[J]. Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering, 2019, 233(3): 653-683
- [4] BEN HADJ R, BELHADJ I, TRIGUI M, et al. Assembly sequences plan generation using features simplification[J]. Advances in Engineering Software, 2018, 119: 1-11.
- [5] ABDULLAH M A, AB RASHID M F F, GHAZALLI Z. Optimization of assembly sequence planning using soft computing approaches: a review[J]. Archives of Computational Methods in Engineering, 2019, 26(2): 461-474.

[6] 郝博,徐东平,王明阳,等.基于改进遗传算法的机翼装配序列智能规划[J].组合机床与自动化加工技术,2021(6):135-137.140

- HAO Bo, XU Dongping, WANG Mingyang, et al. Intelligent planning of aircraft wing assembly sequence based on improved genetic algorithm[J]. Modular Machine Tool & Automatic Manufacturing Technique, 2021(6): 135-137, 140. (in Chinese)
- [7] 黄丰云,熊雄,周铮,等.遗传帝国竞争混合算法在装配序列规划中的研究与应用[J]. 机械设计与制造, 2022(3): 266-271, 275.
 - HUANG Fengyun, XIONG Xiong, ZHOU Zheng, et al. Research and application of hybrid genetic imperial competition algorithms in assembly sequence planning[J]. Machinery Design & Manufacture, 2022(3): 266-271, 275. (in Chinese)
- [8] 王丰产, 孙有朝, 李娜. 多工位装配序列粒子群优化算法[J]. 机械工程学报, 2012, 48(9): 155-162.
 - WANG Fengchan, SUN Youchao, LI Na. Multi Station assembly sequence planning based on particle swarm optimization algorithm [J]. Journal of Mechanical Engineering, 2012, 48(9): 155-162. (in Chinese)
- [9] 刘江伟, 郭宇, 查珊珊, 等. 基于改进粒子群算法的多工位 装配序列规划[J]. 计算机集成制造系统, 2018, 24(11): 2701-2711
 - LIU Jiangwei, GUO Yu, ZHA Shanshan, et al. Multi Station assembly sequence planning based on improved particle swarm optimization algorithm[J]. Computer Integrated Manufacturing Systems, 2018, 24(11): 2701-2711. (in Chinese)
- [10] WANG J F, LIU J H, ZHONG Y F. A novel ant colony algorithm for assembly sequence planning[J]. The International Journal of Advanced Manufacturing Technology, 2005, 25(11): 1137-1143.
- [11] LU Cong, YANG Zhuo. Integrated assembly sequence planning and assembly line balancing with ant colony optimization approach[J]. The International Journal of Advanced Manufacturing Technology, 2016, 83(1): 243-256.

- [12] ZHANG Zaifang, YUAN Baoxun, ZHANG Zhinan. A new discrete double-population firefly algorithm for assembly sequence planning[J]. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 2016, 230(12): 2229-2238.
- [13] MURALI G B, DEEPAK B B V L, BAHUBALENDRUNI M V A R, et al. Optimal assembly sequence planning using hybridized immune-simulated annealing technique[J]. Materials Today: Proceedings, 2017, 4(8): 8313-8322.
- [14] LI Xinyu, QIN Kai, ZENG Bing, et al. Assembly sequence planning based on an improved harmony search algorithm[J]. The International Journal of Advanced Manufacturing Technology, 2016, 84(9): 2367-2380.
- [15] LI Junying, LU Cong. Assembly sequence planning with fireworks algorithm[J]. International Journal of Modeling and Optimization, 2016. 6(3): 195-198.
- [16] GAO Bo, ZHANG Shichao, SUN Hao, et al. Assembly sequence planning based on adaptive gravitational search algorithm[J]. The International Journal of Advanced Manufacturing Technology, 2021, 115(11): 3689-3700.
- [17] YANG Xinshe. Flower pollination algorithm for global optimization[C]//DURAND-LOSE J, JONOSKA N. International Conference on Unconventional Computing and Natural Computation. Berlin, Heidelberg, Germany: Springer, 2012: 240-249.
- [18] MISHRA A, DEB S. Assembly sequence optimization using a flower pollination algorithm-based approach[J]. Journal of Intelligent Manufacturing, 2019, 30(2): 461-482.
- [19] MISHRA A, DEB S. An improved hybrid flower pollination algorithm for assembly sequence optimization[J]. Assembly Automation, 2019, 39(1): 165-185.
- [20] BARNES C J, JARED G E M, SWIFT K G. Decision support for sequence generation in an assembly oriented design environment[J]. Robotics and Computer-Integrated Manufacturing, 2004, 20(4): 289-300

(编辑:秦理曼)